УДК 330:002
ББК 65с51
Н 16
А.В. Нагоев, Н.П. Орлянская

Оптимизация запросов к базе данных в информационной системе учета работы автотранспорта средствами реляционной алгебры

Аннотация:

Предложена методика оптимизации запросов к базе данных информационной системы учета автотранспорта средствами реляционной алгебры.

Ключевые слова:

Учет работы автотранспорта; оптимизация; запросы; базы данных; операции; путевые листы; вход запроса; выход запроса; оболочка запроса; унарные операции; бинарные операции.

В информационных системах учета работы автотранспорта особое внимание уделяется различным выборкам, реализованным путем запросов из файла путевых листов. Именно благодаря созданным запросам производится: ежемесячная обработка путевых листов автомобильного транспорта для учета расхода автозапчастей при ремонте автотранспорта; начисление износа транспортного средства по норме и пробегу, учета взаиморасчетов с клиентами.

Выборка по запросу предшествует формированию статистической отчетности о деятельности автопредприятия: по пробегу автомобилей, по расходу горючего, по сумме начисленной зарплаты водителям по путевым листам. В теории баз данных описывается технология организации запроса к базе данных. Она предусматривает перестановку операций в пределах запроса, идентификацию общих подвыражений и однократном их выполнении, на трансляции и оптимизации запроса относительно фрагментов.

Вместе с тем нельзя не отметить, что в действующих информационных системах запросы в большинстве случаев выполняются в «лоб». Современные средства конфигурирования, проектирования экономических систем дают пользователю в руки инструментарий создания запроса. Формулировка запроса выполняется из соображений удовлетворения пользовательских потребностей, но в них не применяются возможности реляционной аналитики для оптимизации этой операции.

Если обозначить

A – имя некоторого неключевого атрибута;

V – одно из его значений;

e – соответствующее значение первичного ключу;

@ – один из знаков сравнения =; #; <>; <=;>=;

Когда величины V или E найдены, они обозначаются знаком «?»

Можно выделить три класса запросов:

1. Нахождение значения атрибута по известному ключа:

A=(e)=?

2. Нахождение записи с заданным значением неключевого атрибута

A(?)@ v

условия этого типа запросов могут комбинироваться с помощью логических связок «не» «и» «или»

3. Перечислить значения данного атрибута для каждой записи

A(?)=?

Обозначим T – имена атрибутов входа запроса;

t – значение атрибута;

Q – множество атрибутов на выходе запроса.

Входом запроса называется множество имен атрибутов, для которых заданы условия вида A(?)@a; оболочкой запроса называется множество имен атрибутов, упоминаемых в формулировке запроса. Тогда запрос для информационной технологии учета работы автотранспорта можно записать в виде:

Выбрать Q? где T@t

Для оптимизации запроса к реляционной базе данных используются следующие преобразования [1, 2]:

- Объединение операций выборки по нескольким условиям в одну операцию с составным условием;
- Если операция проекции применяется к результату выполнения пересечения, вычитания, или натурального соединения двух отношений, то она может быть предварительно применена к каждому исходному отношению;
- Если операция выборки применяется к результату выполнения пересечения, объединения, вычитания или натурального соединения двух отношений, то она может быть предварительно применима к каждому исходному отношению.

Применим описанную выше технологию для базы данных путевых листов информационной системы учета работы автотранспорта.

Пусть необходимо определить реквизиты клиентов (реквизит Клиент. Организация), которые произвели конкретную группу перевозок (Условно обозначим Группа в таблице ПутевойЛист) за определенный период (Дата), которые разместим в таблице Перевозки. Из построенной ранее инфологической модели Информационной системы учета работы автотранспорта видно что эти две таблицы взаимосвязаны по ключевому полю Код (Код клиента).

Реализацию этого запроса к базе данных с помощью оператора реляционной алгебры можно записать различными способами:

1 способ:

((ПутевойЛист→Код **τ** Клиент →Реквизит) **τ**
(ПутевойЛист → Перевозки=Группа))∩Дата=Реквизит;

где τ – операция полусоединения;

$\cap$ – операция объединения;

Оптимизируем запрос с учетом двух отношений эквивалентности:

1. Каскадность унарных операций позволяет разбить запрос на части, выполняемые последовательно. Унарные операции – это селекция и проекция, уменьшающие размеры отношения по горизонтали и вертикали

R[A1@C1 and A2@C2]≡(R[A1@C1])[A2@C2].

2. Дистрибутивность унарных операций относительно бинарных позволяет распространить их на операнды бинарных операций

(R*S) [A@C] ≡ (R[A@C]*S[A@C];

где R, S отношения,

A – атрибуты,

C – операнды,

@ – оператор сравнения

Известно, [1,2] более раннее применение операций редукции приводит к оптимальному выполнению запроса, его выражение можно записать:

2 способ

(((ПутевойЛист→Перевозки=Группа)∩(ПутевойЛист → Дата=Дата))τ (ПутевойЛист →Код τ Клиент → Реквизит)=Реквизит

Разделим запрос на две части (подзапросы), пусть

Q1=ПутевойЛист → Поездки=Группа;

Q2=ПутевойЛист → Дата=Дата;

Q3=ПутевойЛист → Код ∪ Клиент → Реквизит;

Тогда выражение, обозначенное как 1 способ предлагает выполнение запроса снизу вверх, т.е.

Q3;

Q2;

Q1.

Действительно порядок выполнения операций в выражении запроса слева направо, что соответствует обходу дерева «снизу вверх». Этот способ не является наилучшим с точки зрения затрат на обработку на местах, те выполнения операций в узлах и передачи данных. В этом случае надо передавать атрибут Клиент → Реквизит при выполнении 2 кратных операций полисоединения. Метод сверху вниз соответствует «2 способу» И соответствует выполнению запроса «сверху вниз»

Q1;

Q2;

Q3.

Оба способа приведут к требуемому результату. Однако способ сверху вниз более экономичен, Как с точки зрения времени выполнения, так и объема перемещаемых данных. Это объясняется тем, что мы начинаем с редукции (т.е. селекции), а затем можем использовать полисоединение, для которого можно передавать только значение соединяемых атрибутов.

Произведено распределение селекции относительно естественного соединения и последовательное выполнение проекции на нужные компоненты, а затем распределение их. Проекция начинается теперь сверху, и должна выполняться над атрибутом соединения из соответствующего соотношения, а также над выходным атрибутом Реквизит.

Примечания:

1. Мишенин А.И. Теория экономических информационных систем: Учебник. – 4-е изд., доп. и пераб. – М.: Финансы и статистика, 1999. – 240 с.: ил.
2. Цикритзис Д., Лоховски Ф. Модели данных. – М.: Финансы и статистика, 1985. – 320 с.
3. Орлянская Н.П. Информационная технология бухгалтерского учета. Диссертация на соискание ученой степени канд. техн. наук. – Краснодар, 2001. – 169 с.